DOCUMENT RESUME

ED 230 171 IR 010 686

AUTHOR Caldwell, Robert M.
TITLE Evaluation of Microcomputer Software: How Valid Are the Criteria and Procedures?
PUB DATE Jan 83
NOTE 13p.; Paper presented at the Annual Meeting of the Southwest Educational Research Association (6th, Houston, TX, January 1983).
PUB TYPE Information Analyses (070) -- Viewpoints (120) -- Speeches/Conference Papers (150)

EDRS PRICE MF01/PC01 Plus Postage.
DESCRIPTORS *Computer Programs; *Educational Objectives; *Evaluation Criteria; Evaluation Methods; *Instructional Materials; *Media Selection; *Microcomputers; Qualifications; Research Needs

ABSTRACT

Three issues are critical to the establishment of valid criteria and procedures for determining effective use of computer-based instruction. One issue related to educational software evaluation is that few of the evaluation criteria have been research validated, but, instead, are often based only on speculation and intuition. Criteria are frequently highly inferential, making them highly subjective, which serves to lower reliability among separate ratings. A second problem is the qualifications of individual raters. In a study by Blum (1982), wide variance was found in ratings of software by three or more reviewers, which was attributed to scoring error due to inadequate training or background of the reviewers themselves and subjective judgments of the evaluators on inferential items. A third problem stems from a general lack of knowledge about computer utilization in instruction. Many school districts are acquiring microcomputers without first designing classroom instructional models, and much of the software is used, and therefore reviewed, within possibly inappropriate contexts. Also, most software is purchased without regard to how its objectives fit into existing curricula and overall school goals. (Author/LMM)

****
* Reproductions supplied by EDRS are the best that can be made *
* from the original document. *
****

U.S. DEPARTMENT OF EDUCATION
NATIONAL INSTITUTE OF EDUCATION
EDUCATIONAL RESOURCES INFORMATION CENTER (ERIC)

This document has been reproduced as received from the person or organization originating it.

Minor changes have been made to improve reproduction quality.

Points of view or opinions stated in this document do not necessarily represent official NIE position or policy.

ED230171

EVALUATION OF MICROCOMPUTER SOFTWARE: HOW VALID ARE THE CRITERIA AND PROCEDURES?

Robert M. Caldwell, Ph.D.

The University of Texas Health Science Center

Department of Allied Health Science

5323 Harry Hines Blvd.

Dallas, Texas 75235

(214) 688-2810

Paper presented at the Sixth Annual Meeting of the Southwest Educational Research Association

January 1983

Houston, Texas

IR010686

"PERMISSION TO REPRODUCE THIS MATERIAL HAS BEEN GRANTED BY

Robert M. Caldwell

TO THE EDUCATIONAL RESOURCES INFORMATION CENTER (ERIC)."

ABSTRACT

This paper presents three major issues related to the evaluation of educational software. The first relates to the nature of the criteria used to evaluate software. Few of the criteria used to evaluate software have been validated through research and experimentation; instead, they often find their basis in speculation and intuition only. Criteria for evaluation are frequently highly inferential in nature which, of course, makes them highly subjective. Subsequently, subjectivity can only serves to lower reliability among separate ratings.

A second problem related to the evaluation of software is the qualifications of the individual raters themselves. In a recent study by Blum (1982) wide variance was found in ratings of software by three or more reviewers. This variance was attributed to

A. error in scoring due to inadequate training or background of the reviewers themselves.

B. Sujective judgments of the evaluators on items that were highly inferential.

A third problem in evaluating software stems from a general lack of knowledge about how computers should be used in instruction. The emergence of the inexpensive microcomputer has prompted many school districts to purchase them. Unfortunately, many of these districts are acquiring these micros without first designing classroom models for instruction. The result is that much of the software is used and therefore reviewed within contexts which might be inappropriate. In addition, most software is purchased without regard to how its objectives fit into existing curricula and the overall goals of the school.

These issues, therefore, seem critical to the establishment of valid criteria and procedures for determining effective use of computer-based instruction.

EVALUATION OF MICROCOMPUTER SOFTWARE: HOW VALID ARE THE CRITERIA AND PROCEDURES?

INTRODUCTION

In the past several years there has been a proliferation of software developed to be used with microcomputers in elementary and secondary schools. Unfortunately, much of this software varies greatly in its quality and scope. In an effort to distinguish quality software from "overnight" or "instant" software programs, journals devoted to this new field of "electronic education" have initiated columns where software is reviewed. These reviews typically decribe the purpose of the software, summarize its content, and usually review its perceived effectiveness as a teaching medium. These reviews can serve an important purpose in helping set standards of quality in software development but can also have the deleterious effect of condeming a product that might be useful in certain contexts.

If the product is terrible, then it certainly deserves criticism. Terrible software is usually easy to evaluate. It contains inaccuracies in spelling, content, and structure; is a victim of ineffective lesson design, and suffers from poor screen display and appearance. This type of software,however, represents only a small portion of all the programs available. The major problem with evaluating software programs beyond those which are clearly "terrible" is that it becomes difficult to discriminate between the effective and ineffective ones. Programs that are loaded with graphics and flashy screen displays might not be as

effective as those that are quite simple in appearance and presentation.

In short, the problems inherent in evaluating educational software can be traced to three major issues. The first relates to the nature of the criteria used to evaluate software. Few of the criteria used to evaluate software have been validated through research and experimentation; instead, they Often have their basis in speculation and intuition only. Criteria for evaluation are frequently highly inferential in nature which, of course, makes them highly subjective. Subsequently, subjectivity can only serves to lower reliability among separate ratings.

A second problem related to the evaluation of software is the qualifications of the individual raters themselves. In a recent study by Blum (1982) wide variance was found in ratings of software by three or more reviewers. This variance was attributed to

A. error in scoring due to inadequate training or background of the reviewers themselves.

B. Sujective judgments of the evaluators on items that were highly inferential.

A third problem in evaluating software stems from a general lack of knowledge about how computers should be used in instruction. The emergence of the inexpensive microcomputer has prompted many school districts to purchase them. Unfortunately, many of these districts are acquiring these micros without first designing classroom models for instruction. The result is that much of the software is used and therefore reviewed within

contexts which might be inappropriate. In addition, most software is purchased without regard to how its objectives fit into existing curricula and the overall goals of the school.

These issues, therefore, seem critical to the establishment of valid criteria and procedures for determining effective use of computer-based instruction. In the following paper each of the issues cited above is discussed and recommendations for developing more effective criteria and methods are offered.

EVALUATING EVALUATION GUIDELINES

As more software programs become available, the the number of individuals and organizations who recommend guidelines for evaluating software seems to increase proportionately. As Chairman of the National Council of Teachers of English Committee on Instructional Technology, I have been currently involved in developing a set of criteria for evaluating computer-based materials for the English language arts. In the process, the Committee has examined at least ten (10) separate sets of criteria. In all these sets of criteria, three basic problems seem to emerge. The first problem is that many of the criteria on the evaluation instruments are presented as if they are absolute indicators of program effectiveness when in fact they are based more on conjecture and intuition than on evidence of instructional effectiveness. Few, if any, of these criteria offer any support to show that they are the result of a research effort to establish a relationship between a specific feature of a program and its effect on the achievement of the learner.

Another problem with the criteria we examined was that they

invariably represent each separate criterion as having equal weight in the evaluation of a particular piece of software. None of the guidelines seem to acknowledge that certain features of program design might contribute more to the overall effectiveness of the program than some lesser features.

A final problem was that none of the evaluation guidelines took into account variations in teaching strategy. Instead the criteria were offered to be applied to all software materials regardless if they used a simulation, drill/practice, tutorial, or game format. Logically, one might reason that each of these strategies would utilize the capabilities of the computer in a slightly different way and therefore certain criteria would be less important or not important at all in an evaluation of that strategy.

In short, there are many problems with these guidelines that require more thinking and certainly more research. Taken separately, each of the problems mentioned above might be examined and alternative procedures recommended:

1. VALIDITY OF THE CRITERIA

Probably the most significant problem plaguing the development of evaluation criteria is that, for the most part, few if any of the criteria have been validated in any sort of well conducted research study. We do know from research that some features of an instructional program can affect learner performance, but these are few and are often poorly documented. Most of the criteria included on evaluation guidelines are based on what the developer THINKS contributes to an effective program.

This same problem was inherent in most teacher education research until correlational studies began to establish clear relationships between specific teaching behaviors and student achievement. Unitl these same methods are applied to instruction delivered through a computer-based medium, the problem of validity of criteria will remain a significant one.

One need not look far to find examples to illustrate this point. One criteria which invariably appears on evaluation guidelines is, "Is the program interactive," or "Does the program provide interactions." Few educators would argue that interaction is a valuable characteristic of ANY form of instruction, but it is particularly important when using the full power of the computer-based system. Unfortunately, however, there is wide variablity in the definition of interaction. In a program I evaluated recently the designer posed the question, "In what year was Texas admitted to the Union?" As my response, I was supposed to enter a date. So, as an experiment I entered "2,000,000 B.C." The feedback informed me, "No. Try Again." I then entered, "4,000,456 A.D." The feedback urged me to "Try Again." Finally, I got serious and entered, "1843." The computer told me, "No. Try Again." For some time I really tried to get the right answer; in fact, at one point I just began entering numbers consecutively, "1846," then, "1847," then, "1848," and so on. Each time my effort was rewarded with, "No. Try again."

Now, this program is certainly "interactive" and by the definition implied in most of the evaluation guidelines I have seen, it would receive a positive evaluation on that basis.

However, the program is clearly terrible in that it provides the learners with absolutely no information to help them find the correct answer. Instead, it only serves to increase frustration, inhibit creativity, and stifle the learner.

The validity of other criteria is less subtle. One author offers this criterion, "Does the program offer paging (not scrolling)?" In my opinion, a frame presented in a page format is much more appealing visually than text scrolling up from the bottom of a CRT. But whether that feature contributes more to the effectiveness of a segment of instruction than scrolling does is dubious. But it is this exact type of criteria, that are offered on evaluation guidelines as absolute standards of excellence. As individuals interested in developing and encouraging the development of good software, we must look carefully at these criteria and encourage more validation and research. This can only be accomplished by finding positive correlations between various features of programs and learner achievement.

2. WEIGHTING OF CRITERIA

A second problem which seems to reoccur on evaluation guidelines is that in almost all cases criteria are given equal weight in the evaluation process. That is, each criterion is given equal importance in evaluating the software. Obviously, not all the features of a piece of software contribute equally to the effectiveness of the total program. With this in mind, it would seem logical that evaluation guidelines be structured so that a composite rating put more emphasis on a feature such as, "interactiveness" than on something like, "Feedback is

personalized." Evaluative criteria, therefore, not only needs validation, it also needs to be categorized into a hierarchy which will reflect the degree of contribution each criterion makes to the effectiveness of the overall program.

3. VARIATION IN TEACHING STRATEGY

A final problem seen quite often in evalutive guidelines is that criteria are usually designed to be applied to all software programs regardless of the teaching strategy used to deliver the content. This problem also has an analog in teacher education research. Classroom teaching is very often evaluated without regard for the context of the objectives taught in the lesson. Few educators would argue, however, that very different skills are needed to make individual strategies such as inquiry, lecture, drill, and group discussion effective independently. It would seem to follow, then, that the same criteria applied to evaluating a drill/practice software program might be inappropriate when applying them to evaluate the effectiveness of a computer-based simulation.

QUALIFICATIONS OF REVIEWERS

In the Blum (1982) study a wide variance in ratings of software were found when a single piece of software was reviewed by three or more raters. Blum attributed this variance to two major factors:

1. Sujective judgments of the evaluators on items that were highly inferential, e.g. "the presentation was boring."

This conclusion only serves to verify the need for objective criteria which are validated empirically through correlational

research models. Blum writes, "There is basically little that can be done to reduce variance in evaluation studies when judgements have to be made as they are based on personal opinion and retraining would not alleviate the problem." (p. 27) To a great extent, personal bias will never be eliminated from any evaluation process. However, if we can find and validate objective criteria which rely on low inference (e.g. "Feedback helped shape behavior in the direction of the desired learning outcome.") rather than high inference (e.g. "The program provides adequate feedback."), much of the bias in evaluation that Blum found might be reduced.

2. Error in scoring due to inadequate training or background of the reviewers themselves.

In the Blum study, an evaluation instrument was developed which systematically analyzed software in terms of its intent, contents, methodology and means of evaluation. Unlike many others, this instrument also evaluated the software in terms of the strategy it used.

Blum found that what often occurred was that one rater would respond to the software from the perspective of his/her background while the other would respond from a different background. A mathematics teacher, for example, would look carefully at content in his/her evaluation whereas a rater with instructional design expertise would assess the software from that perspective. Blum's conclusion was that "There is no way to ensure that training and retraining will account for varying background factors so that this (variance) can be reduced." (p. 26)

The solution to this problem seems so obvious that I feel that I have perhaps missed the significance of Blum's conclusion. It would seem that one way to achieve greater reliability among raters would be to have raters from similar backgrounds evaluate the courseware. That is, allow three reviewers who have designed software for the English language arts or for mathematics or for whatever each evaluate the same program. This might provide a certain degree of reliability that is currently lacking.

Another solution to this problem would be to choose raters who had actually designed software so that they had that experience in common as well. Individuals who have tried the laborious and frustrating experience of actually creating software often have a deeper appreciation of what an author or designer was attempting in the program. A final solution might be to have each software program rated by three individuals and then report the composite rating rather than let the review fall on the shoulders of just one reviewer.

LACK OF INSTRUCTIONAL MODELS

A final problem in evaluating software stems from a general lack of knowledge about how software should be used in a classroom. As students we have had the benefit of hundreds of hours of instruction. Each lesson was in itself a model of how (or how not) to teach a lesson. As teachers we emulated the models we found most useful. In using computer-based education, however, few such models exist. Few of us have ever experienced a computer-based course so have little insight into potential problems. As a result, we often evaluate a software program on

the basis of the diskette or computer component itself instead of looking at it as a total system which includes print materials or other media integrated into the system.

These issues, therefore, seem critical to the establishment of valid criteria for evaluating educational software. Tens of thousands of dollars are being invested in software development. It is important that the production of that software is guided by characteristics which have a relationship to learner achievement so that the consumers of that software can make reasoned choices about the software programs they buy.

REFERENCES

Blum, V.L. "Evaluation of Educational Software for Microcomputers: An Analytical Approach," Paper presented at the American Educational Research Association Annual Meeting, New York, 1982.